Report of the data generated with the new VIPIC chip designed at Fermilab

Pietro Peronio[1] and Grzegorz W. Deptuch[2]

[1]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy
[2]Particle Physics Division ASIC Development group, Fermilab, Illinois, USA

September 26, 2014

Contents

1 Introduction

In this paper the results of the analyses made on data acquired with the new VIPIC chip are presented. This chip is a new kind of readout chip meant to be connected to a detector matrix made up of 4096 X-Ray photodetectors. The most innovative element of this chip is the novel 3D-IC which was exploited during its design.

2 Preliminary analyses

The first data that were analyzed had been acquired under *direct beam* condition at Argonne National Laboratory. The matrix was place at the end of a beam line of the synchrotron and it was illuminated through a slit.

The four file that were analyzed are the following:

- `10keV_100MTSCLK_TSin_ns_153nsFrame_NoPhTrigger.vipic`
- `10keV_100MTSCLK_TSin_ns_153nsFrame.vipic`
- `10keV_100MTSCLK_TSin_TSPulse_153nsFrame.vipic`
- `10keV_100MTSCLK_TSin_TSPulse_153nsFrame_TP12.vipic`

The files have an initial header made up of 1028 bytes. The data are organized in 32-bit words, sorted in time-stamp words and pixel-data words. The former ones have the following structure: 1XXXXXXX-XXXXXXXX-XXXXXXXX-XXXXXXXX, that is to say 31 bits of time-stamp and the most significant bit is a header.

The latter ones have the following structure: 010CCCCC-DDDDDDDD-NNNNAAAA-AAAAAAAA, where '010' is the header, C bits are the 5-bit counter bits, D bits are used for debugging, N bits are not used and are '0', A bits are the 12-bit long address bits.

Firstly the time stamps were analyzed. As the headers of the two different 32-bit words are different, it was quite straightforward to sort the time-stamps: the most significant bytes that are greater than or equal to 128 are the right ones.

For each files, all the time-stamps were sorted and the derivative was built in order to see if all of them were equally spaced.

Using the time-stamps and the data, a plot of the number of photons collected in each bunch was built.

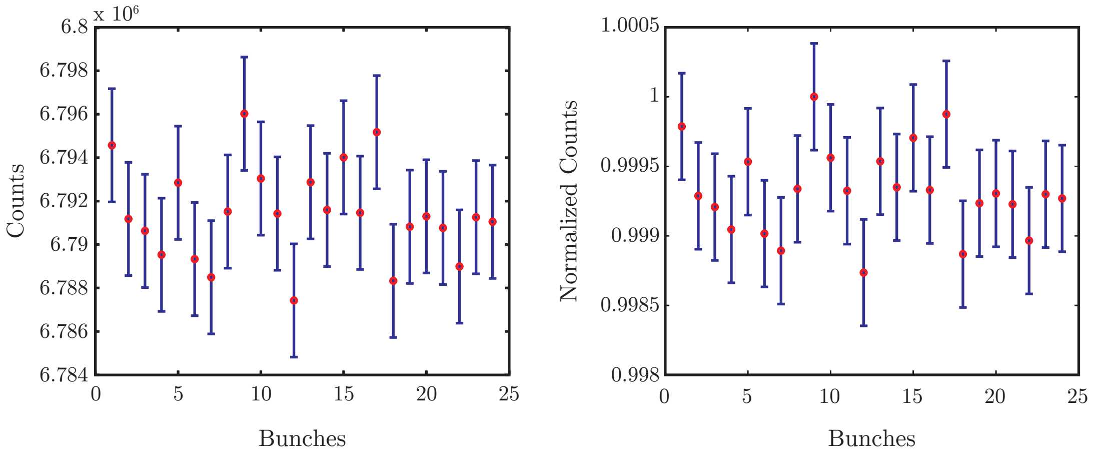
After that a plot of the pixel position versus number of counts was built in order to check if it was consistent with the way in which the photodetector matrix was illuminated.

An histogram of the number of photons collected by each pixel was built afterwards.

2.1 `10keV_100MTSCLK_TSin_ns_153nsFrame_NoPhTrigger`

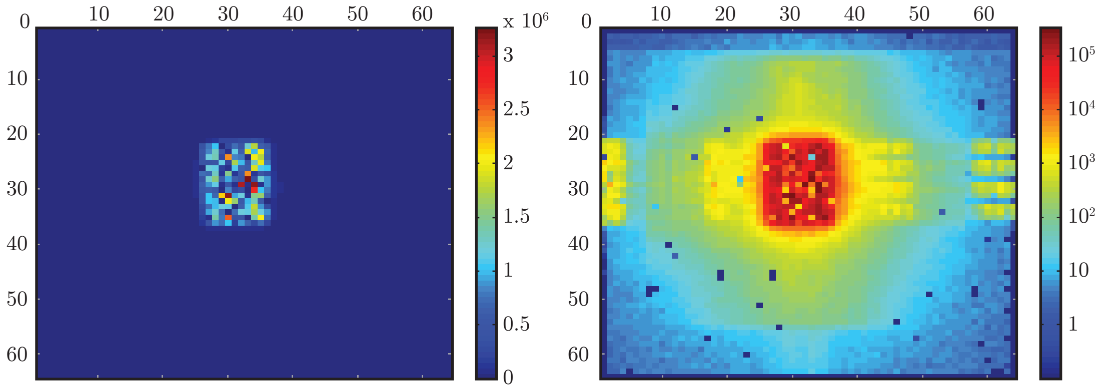
All the time-stamps collected in this file are equally spaced and the numerical distance from two adiacent ones is 22.

In figure 1 the number of photon counts collected in each bunch is represented. As we can see, there is no significant difference between the bunches. This behaviour is consistent with the fact that the readout is not synchronized with the arrival of the photons, so some of these coming in differents bunches can be summed together causing the total number of hits to be averaged on all the bunches.



(a) (b)

Figure 1: Photon counts vs number of bunch: (a) total number of counts, (b) normalized number of counts



(a) (b)

Figure 2: Color map of the pixel use: (a) linear scale, (b) logarithmic scale

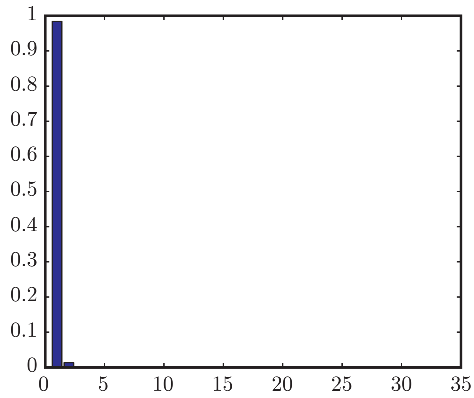


Figure 3: Histogram of the number of photon hits per each period and pixel

In figure 2 the color map representing the total number of photon hits per each pixel is riported.

Because the photodetector matrix was illuminated through a slit, we would expect to see the maximum number of counts in the center and then to see the number counts decreasing moving towards the borders. On the contrary we see an increasing number of counts on the right-central and on the left-central parts of the matrix. Of course this is not physically consistent and it could be related to a problem in data transmission/deserialization which will be further investigated.

It's important to notice that all the 32-bit words have the correct header. As I told before, in order to sort the time-stamps words, those ones which had the most significant byte bigger than or equal to 128 were looked for. Because all the time-stamps I selected in this way were equally spaced, I could infere that their header was correct.

Because the header of the data words is '010', the most significant byte can have a value between 64 (0 counts) and 95 (31 counts). In order to check if the data header was always correct, words whose most significant byte was less than 128 were looked for. For each word selected in that way, it was checked if the the most significant byte had a value between 64 and 95. No wrong word have been found, so all the header of the data words were correct.

Afterwds, the N bits of the third byte in data words, which should be equal to '0', were checked. In order to do that the third byte of each data word was checked to have a value less than or equal to 15. No errors were found.

In figure 3 a histogram of the number of photon hits per each period and pixel is reported. As we can see, each pixels was hit by only one photon per period for most of the time. In table 1 the absolute and realtive frequencies for each number of photon hits are reported. Because of the low intensity of the photons beam, each pixel couldn't have been hit more than once in each period so it's not consistent with physics to have a counter valure bigger than one. A probable cause is that the *clear* signal which is provided to the all counters after each time frame is not working well. Maybe those counters which are not currently readout are keeping on counting so when they are read they will show a bigger value. This is only an assumption based on acquired data. In order to crosscheck it some further investigation should be done using the VIPIC chip.

In figure 4 a color map of the pixel matrix is reported. This time the color map represents the number of occurrences per each pixel of counter value bigger than 2 (figure 4(a)) or less than or equal to 2 (figure 4(b)). This plot was built in order to see

Table 1: Absolute and relative frequency of each number of photon hits

Bin	Absolute frequency	Relative frequency
1	156667120	9.840960e-01
2	2095275	1.316136e-02
3	245384	1.541366e-03
4	73011	4.586146e-04
5	33764	2.120867e-04
6	19208	1.206540e-04
7	12924	8.118141e-05
8	8720	5.477421e-05
9	6787	4.263217e-05
10	5257	3.302156e-05
11	4159	2.612453e-05
12	3442	2.162074e-05
13	2878	1.807800e-05
14	2593	1.628779e-05
15	2142	1.345486e-05
16	1907	1.197872e-05
17	1697	1.065961e-05
18	1448	9.095534e-06
19	1281	8.046532e-06
20	1293	8.121909e-06
21	1138	7.148285e-06
22	995	6.250039e-06
23	899	5.647020e-06
24	886	5.565361e-06
25	829	5.207319e-06
26	747	4.692240e-06
27	721	4.528922e-06
28	677	4.252539e-06
29	649	4.076658e-06
30	598	3.756305e-06
31	588	3.693490e-06

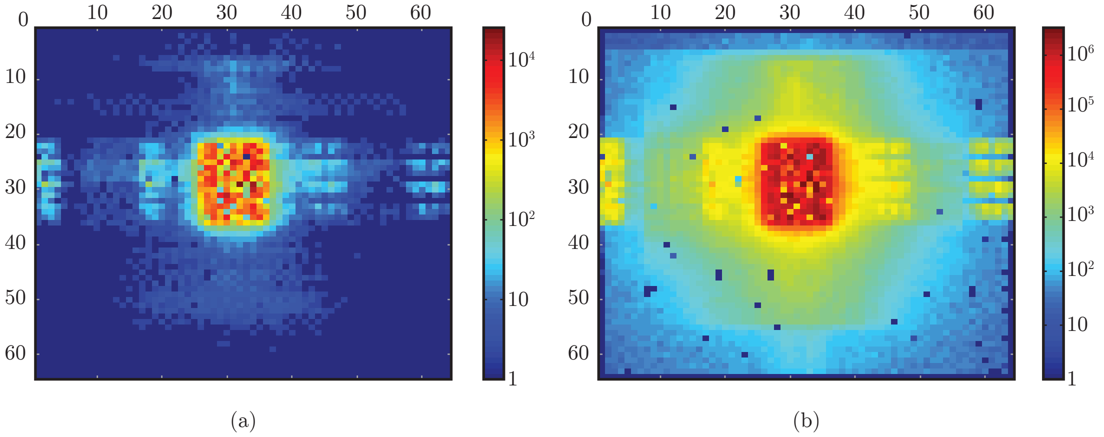


Figure 4: Color map of the occurences per each pixel of counter value: (a) greater than 2; (b) less than or equal to 2

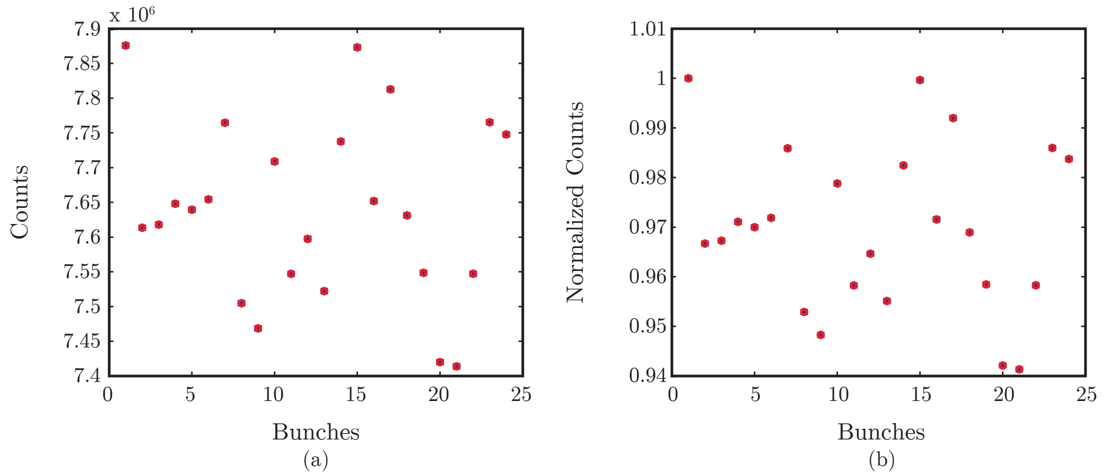


Figure 5: Photon counts vs number of bunch: (a) total number of counts, (b) normalized number of counts

if there was a correlation between the occurence of a counter value greater than 2 and pixel position. It can be inferred that there is no correlation.

2.2 `10keV_100MTSCLK_TSin_ns_153nsFrame`

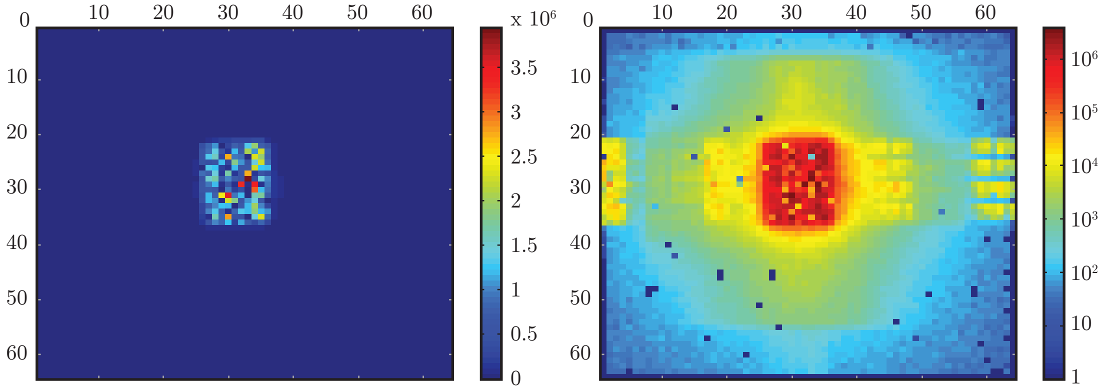
In this file time-stamps are not equally spaced. The numerical distance between the first three time-stamps is 23 and 40. The numerical distance between the further time-stamps is 24 and 25: 44 994 886 times it's 24 and the remaining 55 005 098 times it's 25.

In figure 5 the total number of hits in each bunch is represented. If we compare this plot with figure 1 we can appreciate a bigger diversity among the bunches now. This is due to the fact that an external trigger (APS clock) was used in order to synchronize the acquisition.

In figure 6 the color map representing the total number of photon hits per each pixel is riported.

The error in the number of counts is still present in this plot.

In figure 7 an histogram of the number of photon hits per each period and pixel is



(a) (b)

Figure 6: Color map of the pixel use: (a) linear scale, (b) logarithmic scale

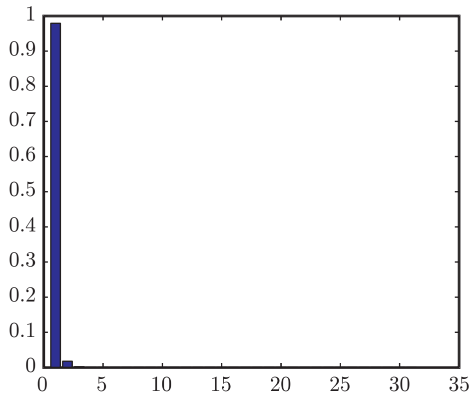


Figure 7: Histogram of the number of photon hits per each period and pixel

Table 2: Absolute and relative frequency of each number of photon hits

Bin	Absolute frequency	Relative frequency
1	174321736	9.791270e-01
2	3122374	1.753769e-02
3	355471	1.996603e-03
4	96320	5.410083e-04
5	42285	2.375056e-04
6	23649	1.328312e-04
7	15485	8.697585e-05
8	10772	6.050396e-05
9	7890	4.431640e-05
10	6221	3.494199e-05
11	4958	2.784800e-05
12	4149	2.330402e-05
13	3393	1.905774e-05
14	3015	1.693459e-05
15	2514	1.412059e-05
16	2199	1.235130e-05
17	1899	1.066627e-05
18	1762	9.896768e-06
19	1471	8.262284e-06
20	1262	7.088377e-06
21	1177	6.610951e-06
22	1089	6.116674e-06
23	946	5.313475e-06
24	877	4.925917e-06
25	901	5.060719e-06
26	809	4.543976e-06
27	658	3.695842e-06
28	700	3.931746e-06
29	713	4.004765e-06
30	598	3.358835e-06
31	635	3.566656e-06

reported. As we can see, each pixels was hit by only one photon per period for most of the time. In table 2 the absolute and realtive frequencies for each number of photon hits are reported.

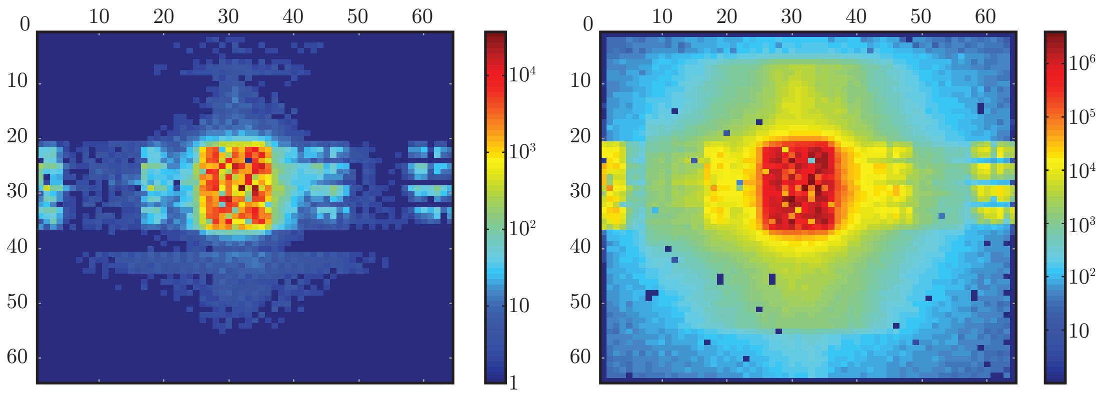
In figure 8 a color map of the pixel matrix is reported. This time the color map represents the number of occurrences per each pixel of counter value bigger than 2 (figure 8(a)) or less than or equal to 2 (figure 8(b)). Also for this file there is no correlation.

2.3 `10keV_100MTSCLK_TSin_TSPulse_153nsFrame`

In this file the time-stamps are all equally spaced and the numerical distance is 1.

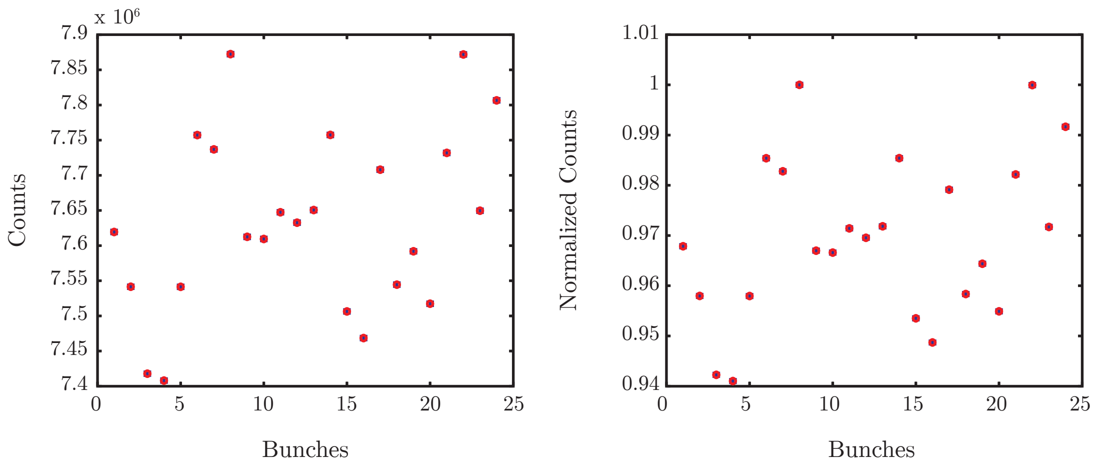
In figure 9 the total number of counts in each bunch is reported.

In figure 10 the color map representing the total number of photon hits per each pixel is riported.



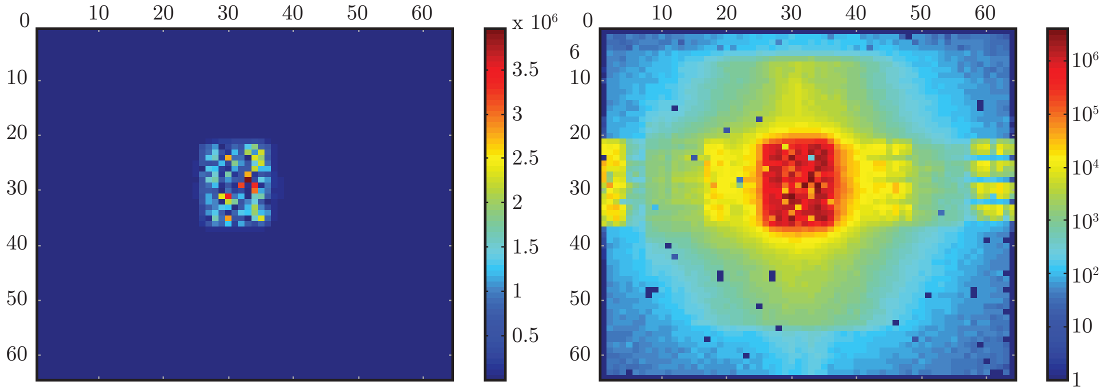
(a) (b)

Figure 8: Color map of the occurences per each pixel of counter value: (a) greater than 2; (b) less than or equal to 2



(a) (b)

Figure 9: Photon counts vs number of bunch: (a) total number of counts, (b) normalized number of counts



(a) (b)

Figure 10: Color map of the pixel use: (a) linear scale, (b) logarithmic scale

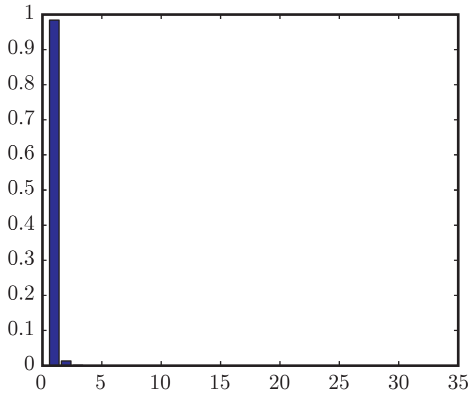


Figure 11: Histogram of the number of photon hits per each period and pixel

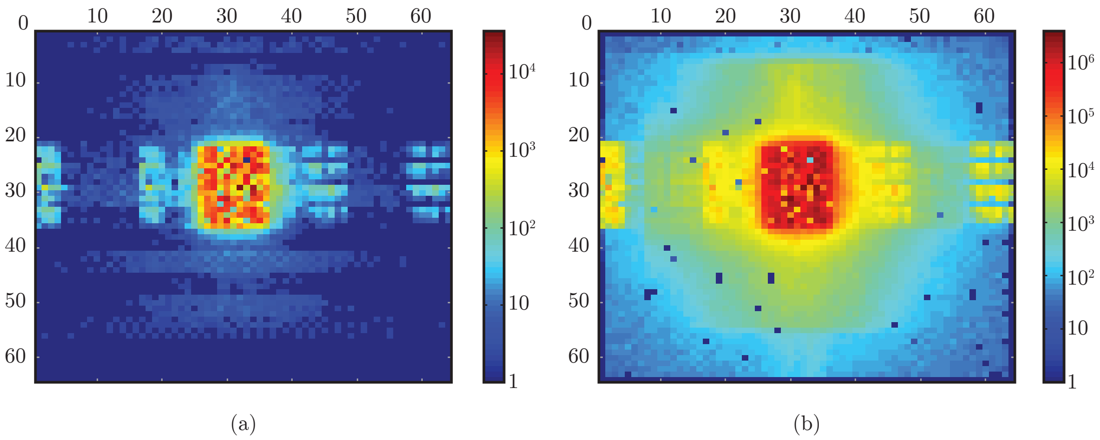


Figure 12: Color map of the occurences per each pixel of a counter value: (a) greater than 2; (b) less than or equal to 2

The error in the number of counts is still present in this plot.

In figure 11 an histogram of the number of photon hits per each period and pixel is reported. As we can see, each pixels was hit by only one photon per period for most of the time. In table 3 the absolute and realtive frequencies for each number of photon hits are reported.

In figure 12 a color map of the pixel matrix is reported. This time the color map represents the number of occurrences per each pixel of counter value bigger than 2 (figure 12(a)) or less than or equal to 2 (figure 12(b)). Also for this file there is no correlation.

2.4 `10keV_100MTSCLK_TSin_TSPulse_153nsFrame_TP12`

In this file the time-stamps are all equally spaced and the numerical distance is 1.

In figure 13 the total number of counts in each bunch is reported.

In figure 14 the color map representing the total number of photon hits per each pixel is riported.

The error in the number of counts is still present in this plot.

In figure 15 an histogram of the number of photon hits per each period and pixel is reported. As we can see, each pixels was hit by only one photon per period for most of the time. In table 4 the absolute and realtive frequencies for each number of photon

Table 3: Absolute and relative frequency of each number of photon hits

Bin	Absolute frequency	Relative frequency
1	174237464	9.792385e-01
2	3098509	1.741405e-02
3	353552	1.987011e-03
4	96627	5.430570e-04
5	42762	2.403283e-04
6	24238	1.362209e-04
7	16152	9.077646e-05
8	11261	6.328837e-05
9	8468	4.759132e-05
10	6556	3.684562e-05
11	5189	2.916289e-05
12	4109	2.309314e-05
13	3390	1.905227e-05
14	2959	1.662999e-05
15	2605	1.464046e-05
16	2213	1.243736e-05
17	1848	1.038601e-05
18	1661	9.335048e-06
19	1526	8.576330e-06
20	1349	7.581565e-06
21	1228	6.901529e-06
22	1025	5.760641e-06
23	1085	6.097849e-06
24	884	4.968201e-06
25	846	4.754636e-06
26	812	4.563552e-06
27	776	4.361227e-06
28	653	3.669950e-06
29	619	3.478865e-06
30	604	3.394563e-06
31	625	3.512586e-06

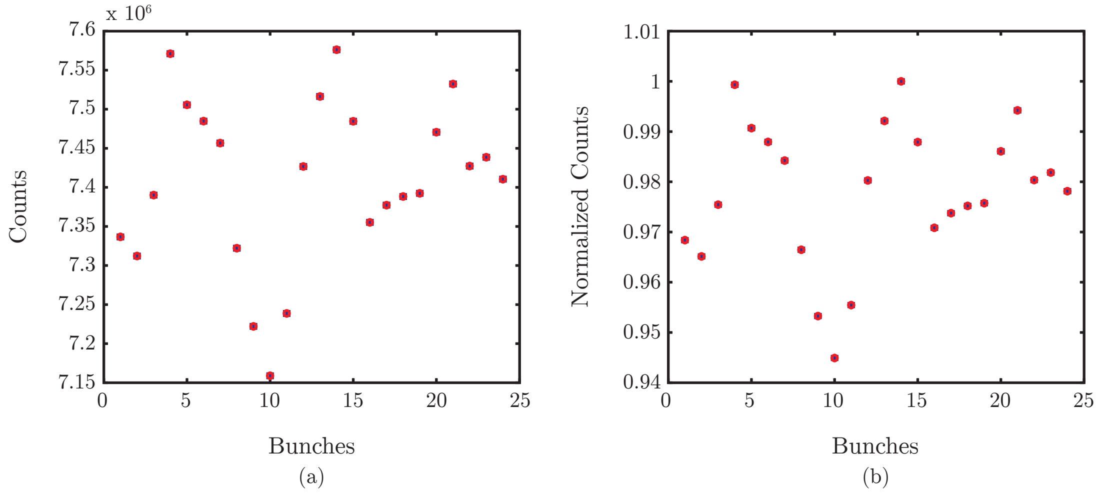


Figure 13: Photon counts vs number of bunch: (a) total number of counts, (b) normalized number of counts

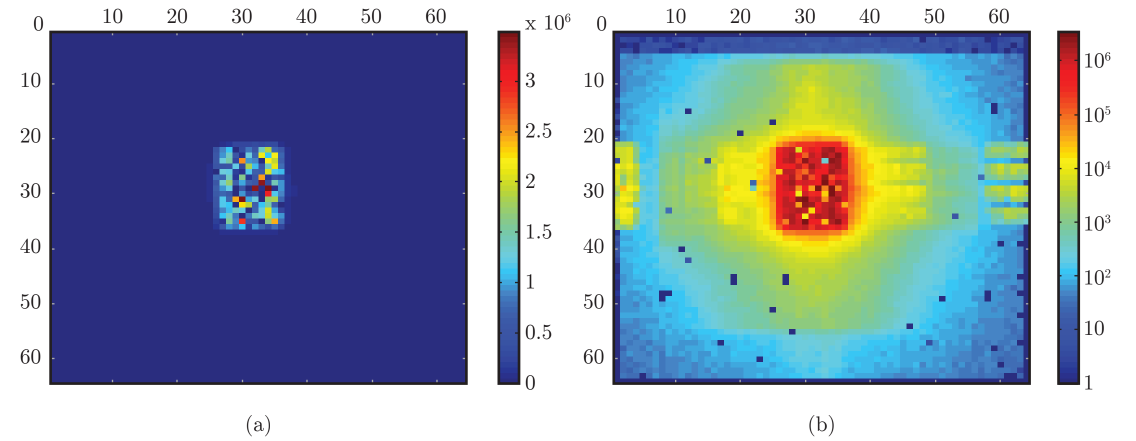


Figure 14: Color map of the pixel use: (a) linear scale, (b) logarithmic scale

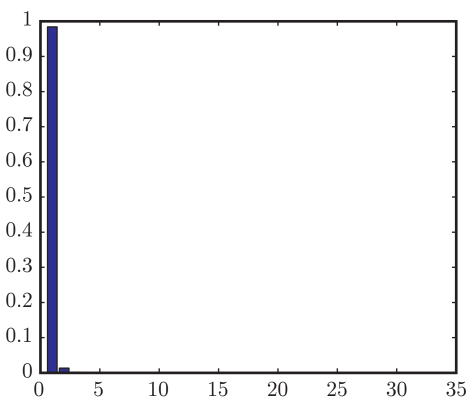


Figure 15: Histogram of the number of photon hits per each period and pixel

Table 4: Absolute and relative frequency of each number of photon hits

Bin	Absolute frequency	Relative frequency
1	171620207	9.857032e-01
2	2065413	1.186273e-02
3	242489	1.392739e-03
4	71827	4.125394e-04
5	32002	1.838040e-04
6	18218	1.046354e-04
7	11697	6.718189e-05
8	7910	4.543120e-05
9	5880	3.377187e-05
10	4657	2.674755e-05
11	3695	2.122229e-05
12	3126	1.795423e-05
13	2616	1.502503e-05
14	2158	1.239450e-05
15	1903	1.092991e-05
16	1675	9.620387e-06
17	1444	8.293635e-06
18	1423	8.173021e-06
19	1231	7.070267e-06
20	1128	6.478685e-06
21	1088	6.248944e-06
22	1037	5.956025e-06
23	919	5.278290e-06
24	841	4.830296e-06
25	776	4.456967e-06
26	800	4.594812e-06
27	692	3.974512e-06
28	662	3.802207e-06
29	666	3.825181e-06
30	640	3.675849e-06
31	598	3.434622e-06

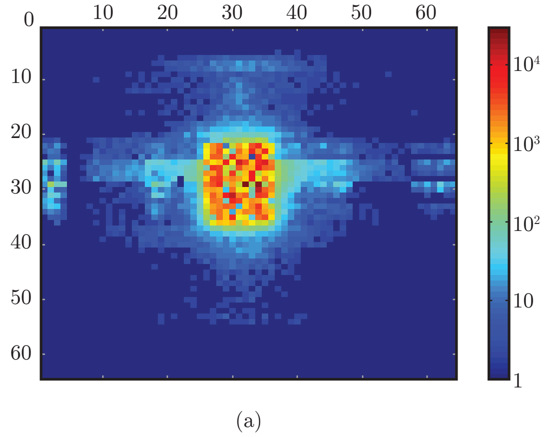
hits are reported.

In figure 16 a color map of the pixel matrix is reported. This time the color map represents the number of occurrences per each pixel of counter value bigger than 2 (figure 16(a)) or less than or equal to 2 (figure 16(b)). Also for this file there is no correlation.

3 Autocorrealtion analyses

One of the main purposes for which the VIPIC chip has been designed is the *X-Ray Photon Correlation Spectroscopy* (XPCS). By means of this technique the dynamic properties of a sample can be inferred from the shape of the autocorrelation function of the measured light intensity.

In this application, the VIPIC chip has been used to collect the photon hits and so



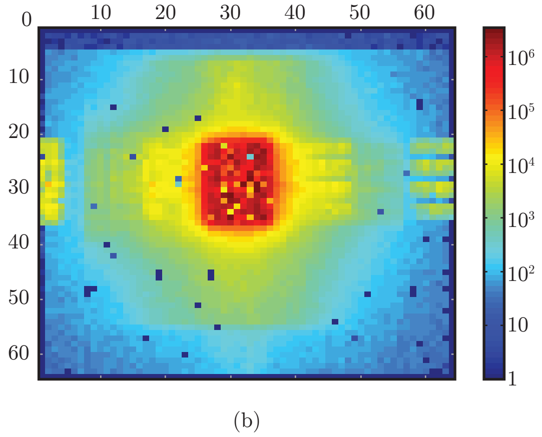


Figure 16: Color map of the occurences per each pixel of counter value: (a) greater than 2; (b) less than or equal to 2

the light intensity was evaluated from the acquired data. Afterwards the autocorrelation function was calculated running a Matlab software provided by researchers from Argonne National Laboratory.

The first autocorrelation analysis was done on one of the files that had been previously analyzed, that is to say on data acquired under a *direct beam* lighting condition.

3.1 10keV_100MTSCLK_TSin_TSPulse_153nsFrame

First the autocorrelation function was calculated and some pixel were analyzed in order to see if there were some differences.

The number of photons that hit each pixel in a given time frame follows a Poissonian distribution. Let's consider $x_{p_i}[n]$ the temporal sequence of the total number of photon hits per frame in pixel p_i, where n represents the number of observed frame. Let's call μ the mean number of photon hits per frame. This mean number is the same for all the frames because they have the same 153 ns time width.

By definition the autocorrelation is:

$$R(n, n+m) = E[x_{p_i}[n] \cdot x_{p_i}[n+m]] \tag{1}$$

Due to the Possonian distribution of photon hits, the numbers of events collected in two differents time frames are independent, so:

$$R(n, n+m) = R(m) = \begin{cases} E[x_{p_i}[n]] \cdot E[x_{p_i}[n+m]] = \mu^2 & \text{if m>0} \\ E[x_{p_i}[n]^2] = \sigma^2 + E[x_{p_i}[n]]^2 = \mu + \mu^2 & \text{if m=0} \end{cases} \tag{2}$$

where σ^2 is the variance of $x_{p_i}[n]$.

The function that is commonly used in this kind of analyses is a normalized autocorrelation function:

$$g^{(2)}(p_i, \tau) = \frac{< I(p_i, t) \cdot I(p_i, t+\tau) >_t}{< I(p_i, t) >_t^2} \tag{3}$$

where $I(t)$ is the light intensity, $< \cdots >_t$ represents a time average over all the equivalent times t and τ is the time delay.

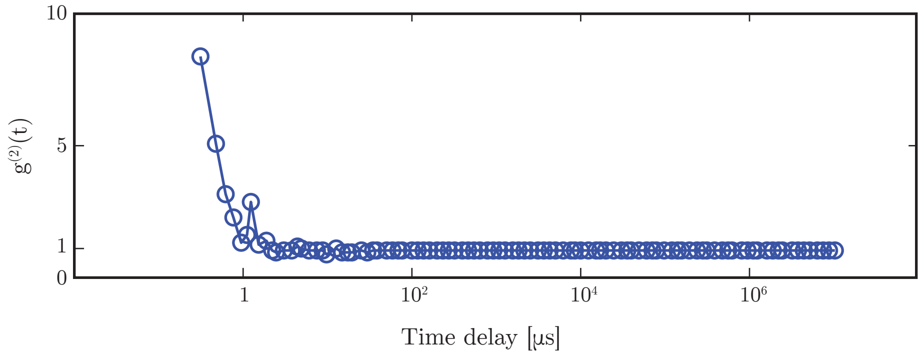


Figure 17: $g^{(2)}$ function for pixel #1369. A flat shape is observed only for time delays higher than 1 µs

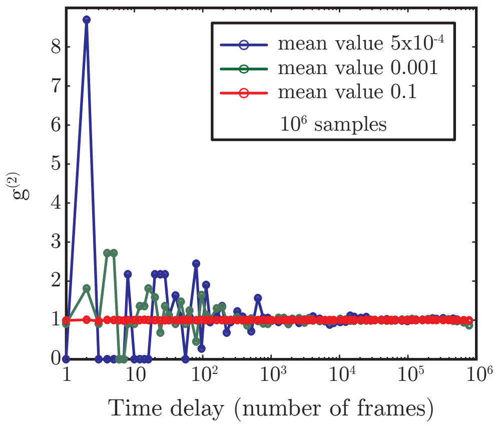


Figure 18: Autocorrelation for number sequences generated by Poissonian processes with different mean values. Increasing the mean value will cause the fluctuation at short time delays to disappear

So, considering $x_{p_i}[n]$, we get:

$$g^{(2)}(p_i, m) = \frac{R(m)}{E[x_{p_i}[n]]^2} = \begin{cases} 1 & \text{if m>0} \\ 1 + \frac{1}{\mu} & \text{if m=0} \end{cases} \tag{4}$$

Looking at the $g^{(2)}$ function flat shape is expected for $m > 1$.

The autocorrelation function is quite similar for all the pixels so in figure 17 the calculated $g^{(2)}$ function only for pixel number 1369 is reported. The shape of the function is flat and equal to 1 only for higher time delays.

In a first hypothesis this shape was due to the electronics of the VIPIC chip. When a photon hits a pixel, an analog pulse with a duration of several hundreds of micro seconds is generated. Because of the very short time frame duration (153 ns) this pulse covers more than one period, causing that pixel to be blind to photon hits in the subsequent time frames. This mechanism can affect the statistics of collected photons.

In order to investigate this hypothesis a fake $x[n]$ sequence that follows a Poissonian distribution was created with Matlab. Before any other test, the statistics of the created sequence were analyzed in order to verify that the process was Poissonian. Different sequences with different mean values but always made up of one million of elements were created and their autocorrelations were compared.

As you can see in figure 18, there is a strong dependance between the mean value

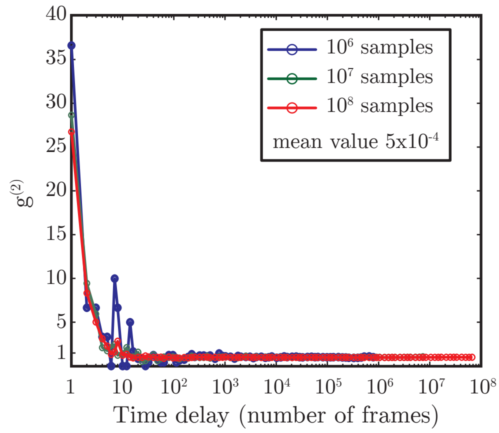


Figure 19: Autocorrelation evaluated for pixel 1369 for different number of total frames

of the sequence and the fluctuation that happens at short time delays. This could be explained by the finite length of the sequences. As shown in equation 4, the autocorrelation function has a peak at time delay equal to zero, whose value is $g^{(2)}(0) = 1+1/m$. If we had an infinite length sequence the evaluated autocorrelation function would fit perfectly the $g^{(2)}$ function. As we are managing finite length sequences I would expect the autocorrelation function not to drop abruptly from $g^{(2)}(0)$ to 1 but to have a peak with a finite time width. If we increase the mean value, we will decrease $g^{(2)}(0)$ and the ringings too.

In figure 19 the $g^{(2)}$ function was evaluated for pixel 1369 changing the total length of the sequence, that is to say taking into account a different number of frames. As we can see, the bouncings at short time delays decrease as the total number of frames is increased.

3.2 `25C_10keV_3.682usFrame_Silica20p_PhTrig_5MTSCLK_Th430`

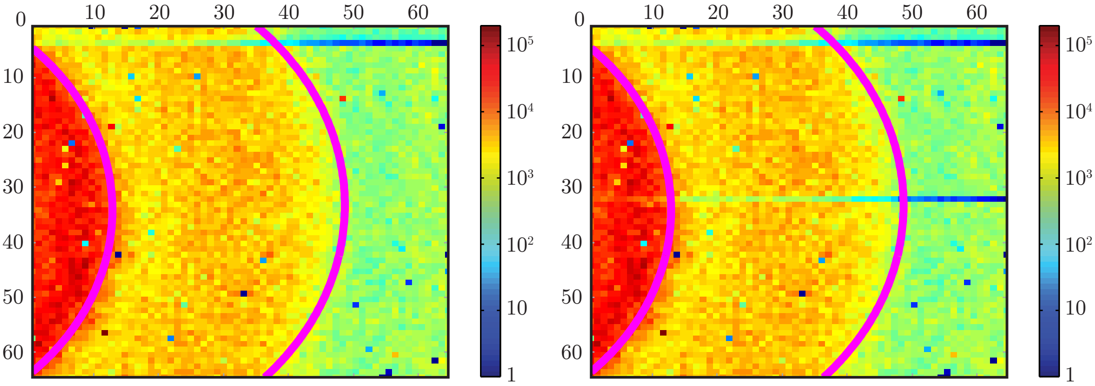
After those preliminary test data taken illuminating a real colloid sample were analyzed.

In this file there were some corrupted data: some 32-bit data had a wrong header. Instead of having the classic '010' header, they had a '11X' one. This caused the software to misunderstand those data and consider them as time stamps. Indeed, reading the sequence of time stamps it was found that there were wrong data inserted in the middle of the sequence (i.e. instead of having a sequence made up of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 there was a sequence of 1, 2, 1092683739, 3, 1094715352, 4, 1096812428, 5, 1098844121, 6, 1100941280, 7).

In figure 20 the color map of the total number of events per each pixel is reported. Two plots have been made in order to understand if wrong data need to be taken into account or not. By looking at the uniformity of the plots it can be inferred that wrong data have only a wrong header whereas the information related to pixels address and photon counts is correct. In figure 20 (b) some rows in the central part of the matrix seems to have been missed.

In figure 21 there is a representation of the addresses contained in the wrong data. In particular, the color map shows the number of times that each address was found in a wrong data.

In figure 22 the same information is reported in a different way. As it can bee noticed, only 256 addresses are contained in wrong data, in particular addresses from 1793 to 2048, which are those connected to readout group number 8.



(a) (b)

Figure 20: Color map of the total number of photons collected by each pixel: (a) taking into account wrong data, (b) discarding wrong data. The purple line delimits the areas with different light intensity. In figure (b) some rows in the central part of the matrix seems to have been missed

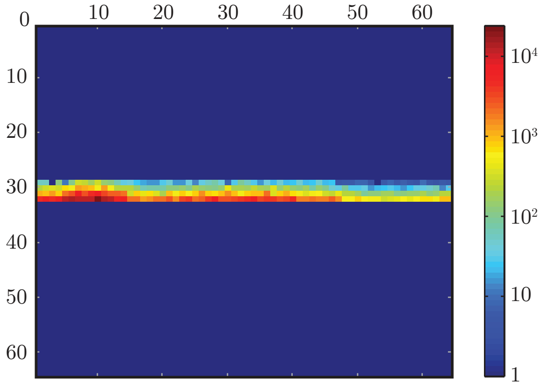
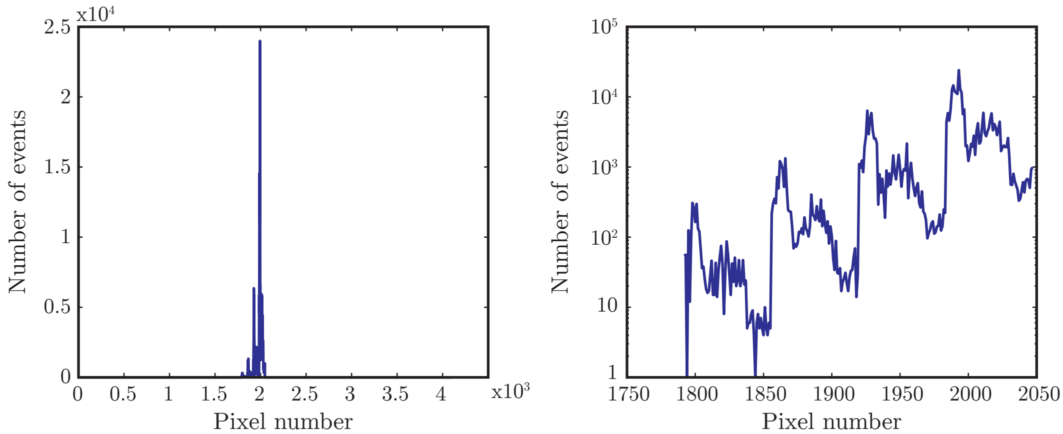
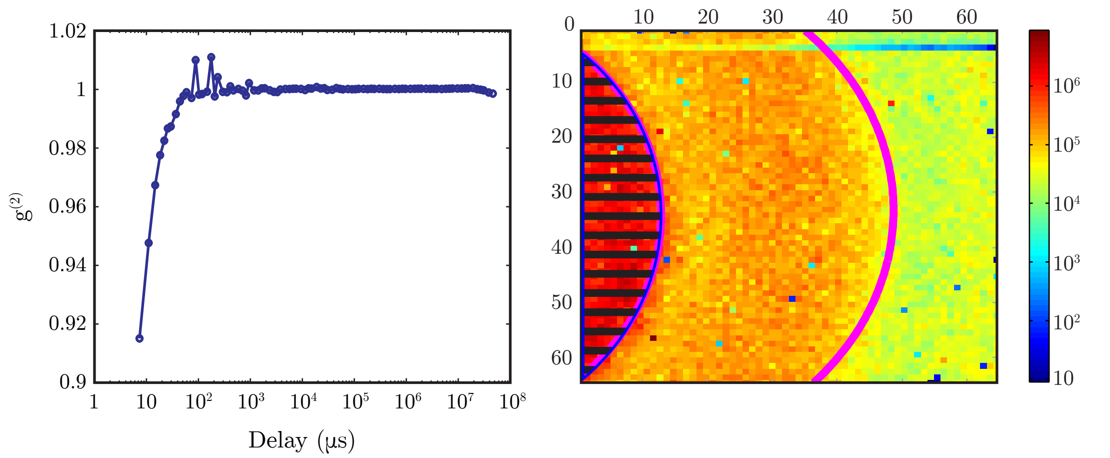


Figure 21: Color map created considering the addresses contained in the wrong data. For each pixel the number of times that its address was found in a wrong data is represented. The plot highlighted that the errors were related only to a group of pixels



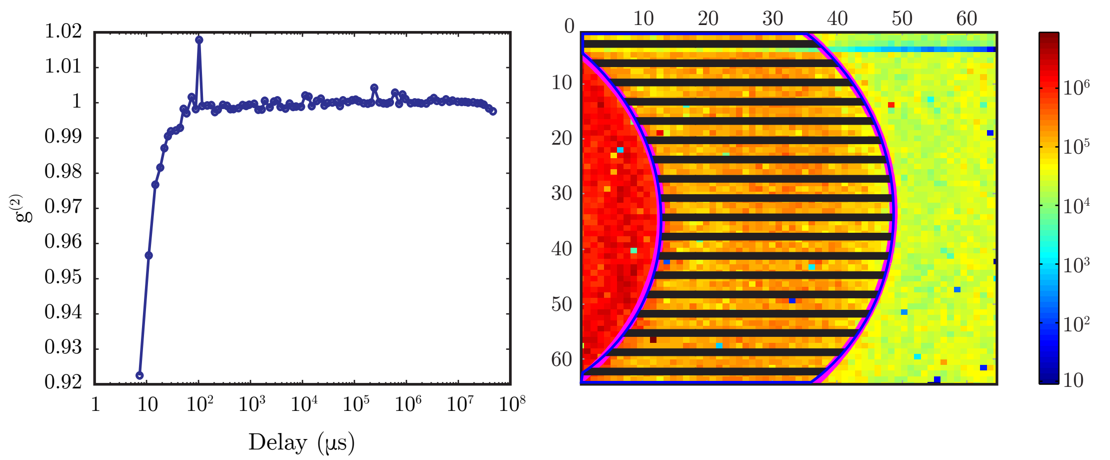
(a) (b)

Figure 22: Number of times that a pixel address was found in a wrong data: (a) linear scale, (b) log scale



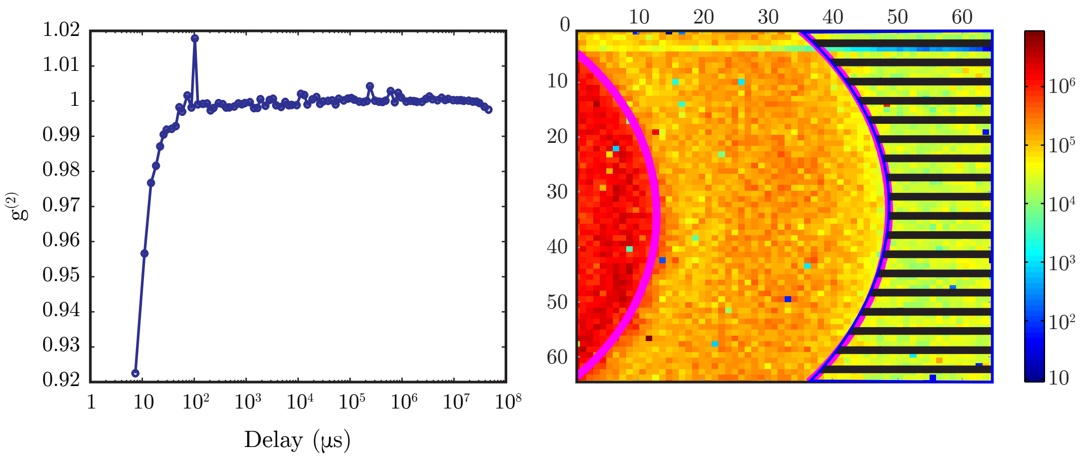
(a) (b)

Figure 23: (a) $g^{(2)}$ function averaged on the pixels of the first anular ring. (b) color map of the total number of counts of the matrix with the first anular ring highlighted



(a) (b)

Figure 24: (a) $g^{(2)}$ function averaged on the pixels of the second anular ring. (b) color map of the total number of counts of the matrix with the second anular ring highlighted



(a) (b)

Figure 25: (a) $g^{(2)}$ function averaged on the pixels of the third anular ring. (b) color map of the total number of counts of the matrix with the third anular ring highlighted

The $g^{(2)}$ functions were then calculated. In order to reduce the variance of the measurement, the autocorrelation functions of the pixels that belong to the same "anular ring" were averaged. The results are reported in figures 23, 24 and 25.

Discarding the very first time delays, the shape of these curves is flat as if they were poissonian. The reason why the shape is flat could be related to the average that have been done. We expect not to see a poissonian distribution because of the dynamics of the colloid which affect photon statistics. If we consider all photons hitting the matrix, of course statistics will be poissonian, but if we focus on a small amount of pixel, we should be able to see different statistics.